Energy, Mines and Resources Canada | Énergie, Mines et Ressources Canada

CANMET

Canada Centre for Mineral and Energy Technology | Centre canadien de la technologie des minéraux et de l'énergie

A SYSTEM FOR GATHERING DRAGLINE DATA USING AN ELECTRONIC DATA LOGGER AND MODEM

N.J. Stuart and P.D. Guyn
Calgary Coal Research Laboratory

December 1984

ENERGY RESEARCH PROGRAM
COAL RESEARCH LABORATORIES
REPORT ERP/CRL 84-44(TR)

MICROMEDIA
FEB 27 1985

A SYSTEM FOR GATHERING DRAGLINE DATA USING AN ELECTRONIC DATA LOGGER AND MODEM

by

N.J. Stuart[1] and P.D. Guyn[2]

SUMMARY

To facilitate operations research into surface mining methods, time study data on draglines was re ed. This study details the development of a system that records events by manual entry on a purpose built keyboard. The data is stored in a microprocessor based data logger and transmitted daily to the laboratory computer via an acoustic modem. Programs were written to analyse the data files and produce statistical summaries outlining the relationships between parameters such as the dragline swing angle and the expected time to complete an operation (e.g. mean swing time).

Copious details of the operation are given and it is hoped that the report can serve as a manual to future users. Details are given of future improvements that could be made to aid data transmission and ease of use.

[1]Research Scientist, [2]Summer Student, Calgary Coal Research Laboratory, CANMET, Energy, Mines & Resources, Calgary, Alberta

SYSTÈME UTILISANT UN ENREGISTREUR ÉLECTRONIQUE ET UN MODEM POUR LA CUEILLETTE DE DONNÉES SUR LA PELLE À BENNE TRAÎNANTE

par

N.J. Stuart[1] et P.D. Guyn[2]

SOMMAIRE

Afin de faciliter la recherche sur les méthodes de fonctionnement des exploitations à ciel ouvert, il était nécessaire d'obtenir des données sur l'utilisation du temps dans les manoeuvres requérant l'emploi de la pelle à benne traînante. Ce rapport présente le détail de la mise en oeuvre d'un système qui enregistre les événements par introduction manuelle des données sur un clavier construit à dessein. Celles-ci sont emmagasinées par un enregistreur electronique articulé sur microprocesseur et transmises quotidiennement à l'unité de calcul du laboratoire par un système modulateur-démodulateur acoustique. La programmation a été faite en vue de l'analyse des fichiers de données et de la production de résumés statistiques retraçant les rapports entre les paramètres tels l'angle de rotation de la pelle à benne traînante et le délai prévu pour l'accomplissement d'une rotation.

Ce rapport présente des détails abondants sur le fonctionnement du système et on espère que celui-ci pourra servir de guide aux futurs usagers. De plus, le rapport contient des suggestions précises quant aux améliorations possibles visant à faciliter la transmission des données et l'utilisation du système.

---

[1]Chercheur scientifique, [2]Étudiant, emploi d'été, Laboratoire de recherche sur le charbon de Calgary, CANMET, Énergie, Mines et Ressources, Calgary, Alberta.

CONTENTS

	Page
SUMMARY	ii
INTRODUCTION	1
DATA ENTRY	1
THE POLYCORDER PROGRAM	2
OPERATING THE POLYCORDER	4
TRANSFERRING DATA	4
PROCESSING THE DATA	6
BAD CHOICES - FUTURE IMPROVEMENTS	6
ADVANTAGES OF THIS SYSTEM	7
APPENDIX A: COMMUNICATIONS PARAMETERS	A-8
APPENDIX B: THE AUXILIARY KEYPAD KEY DESIGNATIONS	B-9
APPENDIX C: THE DATABASE-INITIAL COMPUTATIONS	C-10
APPENDIX D: FORTRAN PROGRAM "U" FOR TRANSFERRING DATA ONTO VAX	D-19
APPENDIX E: POLYCODE PROGRAM FOR DATA ACQUISITION	E-20
APPENDIX F: THE DAILY FIELD ROUTINE	F-21
APPENDIX G: SUMMARY OF DATA-THE INITIAL ANALYSIS	G-22

INTRODUCTION

CANMET's studies of operations in open pit coal mines require a large accurate data base of time study measurements on which to base modelling work. Such work is aimed at various aspects of stochastic processes and optimization and is discussed elsewhere (V.Srajer;"Evaluation of Truck/Shovel System at Highvale Mine"CANMET/CRL Report 1984). This report represents the first steps in computerising data gathering field work, reducing the need for stopwatch and notepad.

The hardware mentioned in this report includes a specially constructed data entry keypad, a polycorder, an acoustically coupled modem for use in the field and a VAX-11/750 computer (with a directly connected modem) which receives and processes the data. Polycorders are products of Omnidata. They are microprocessor based data logging devices, the model in question having a 16K RAM memory with a rechargeable and a backup battery. VAX computers are a product of Digital. They are 32 bit minicomputers.

The software described in this report includes the polycroder keyboard reading and storage program, the VAX's Fortran program for receiving data, Fortran programs for initial interpretation of this data. Also the settings and procedures necessary for the above to work are explained and described. This hardware and software and these procedures and settings are as used to monitor dragline operation at Highvale Mine, Alberta in July/August 1984. All programs described are entirely original, and are solely the work of the authors.

DATA ENTRY

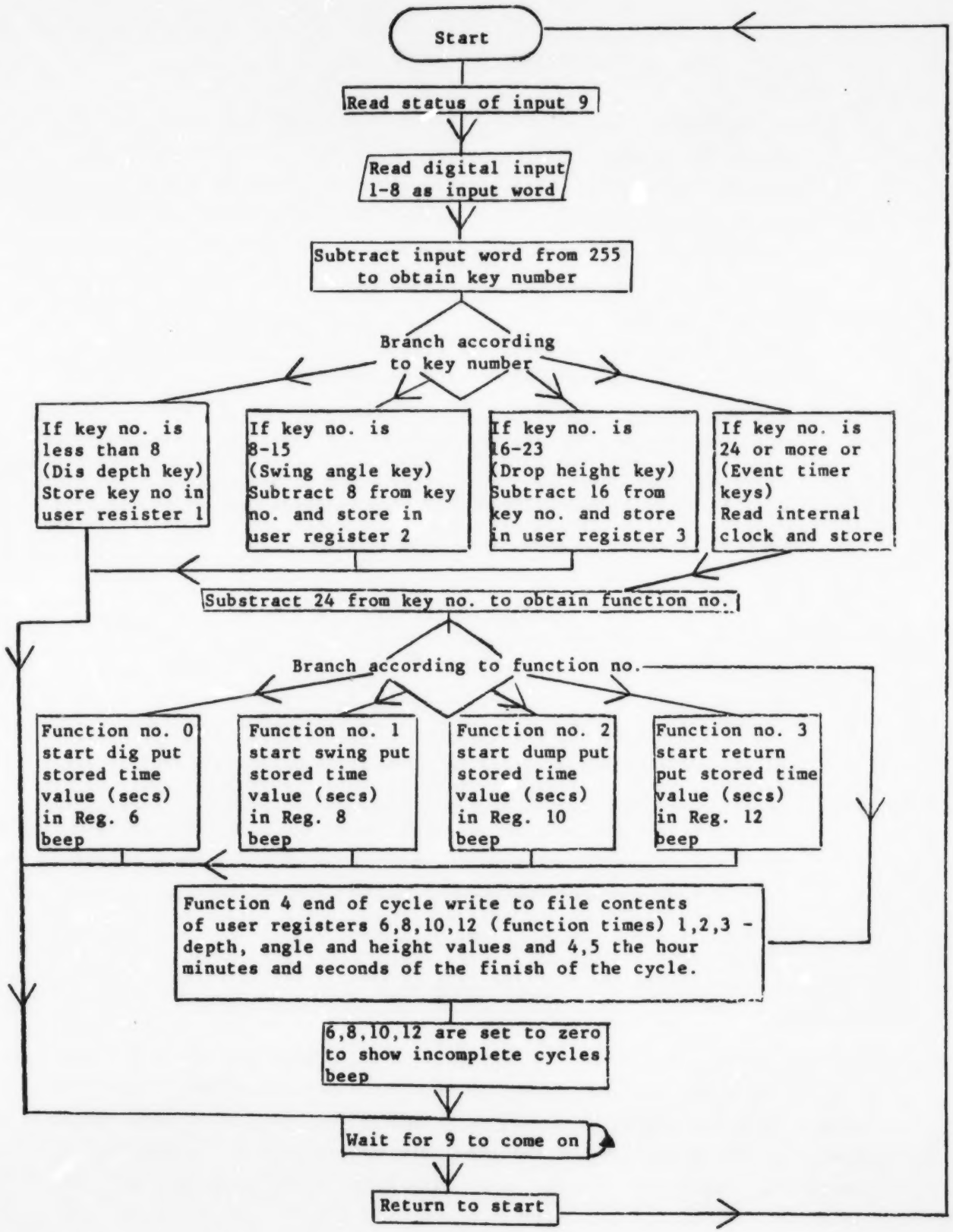
A 32 key data entry module was constructed from a suitable number of switches and a network of diodes. This module plugs into the 9 digital inputs of the polycorder. In the state in which no keys are pressed all channels are in the "ON" state. Keys switch off unique combinations of inputs by connection through a suitable resistor (350 ohm) and diodes to the polycorder ground. Each key causes channel 9 to go off - this is a special input. The other channels can be read as a single number. When this input number is subtracted from 255 the result is the "key number". These are

0-31 (See Appendix B). The functions of individual keys are thus defined by the software in the polycorder and the keypad thus has a wide variety of potential applications.

The keys are arranged in 4 rows of 8. The top row has been designated the "dis depth" row. Each key corresponds to an estimated dig depth of the dragline. The exact values represented by each button are to be selected in the field so that they cover a reasonable range. The second and third rows are designated the "swing angle" and "dump height" rows which similarly correspond to suitable values for field observation. The last row is the event marker row. The first 5 keys only are defined: "start digging", "start swinging", "start dump", "start return" and "end of cycle. These keys cause time values to be stored. Additionally the "end of cycle" causes the time values for the entire cycle to be put into file together with the numbers of the "dig depth", "swing angle" and "dump height" buttons last pushed. Each cycle uses 15 bytes of polycorder memory.

THE POLYCORDER PROGRAM

The polycorder must be programmed in "polycode". This resembles machine language - operations being rudimentary and performed entirely in one of two stacks. The program currently used consists of 114 lines of code and may be summarized as follows:


Start
Read status of input 9
Read digital input 1-8 as input word
Subtract input word from 255 to obtain key number
Branch according to key number
If key no. is less than 8 (Dis depth key) Store key no in user resister 1
If key no. is 8-15 (Swing angle key) Subtract 8 from key no. and store in user register 2
If key no. is 16-23 (Drop height key) Subtract 16 from key no. and store in user register 3
If key no. is 24 or more or (Event timer keys) Read internal clock and store
Substract 24 from key no. to obtain function no.
Branch according to function no.
Function no. 0 start dig put stored time value (secs) in Reg. 6 beep
Function no. 1 start swing put stored time value (secs) in Reg. 8 beep
Function no. 2 start dump put stored time value (secs) in Reg. 10 beep
Function no. 3 start return put stored time value (secs) in Reg. 12 beep
Function 4 end of cycle write to file contents of user registers 6,8,10,12 (function times) 1,2,3 - depth, angle and height values and 4,5 the hour minutes and seconds of the finish of the cycle.
6,8,10,12 are set to zero to show incomplete cycles beep
Wait for 9 to come on
Return to start

OPERATING THE POLYCORDER

The polycorder has a 21 key control pad using membrane switches. The following are important to this system:

- Mode 9 - sleep mode. This in this mode the display is blank but the values in the registers are maintained. Sleep mode is left by pressing the "ON" button. Future modes including sleep (mode 9) are obtained by pressing the appropriate number button when the prompt "mode ?" appears.

  NOTE: all other modes can be left by hitting the "esc" button.
- Mode 0 - This mode executes polycode programs, entering this mode one selects the file to be run by using the left and right facing arrows and then hits the "ENTER" button. The keyboard program is called "drag.p".
- Mode 1 - this is the mode used to transmit the data after mode 7:1 has been used to run an aquisition program on the VAX. Mode 2 options 1 or 2 may be used.
- Mode 2 - Option 3 of this mode is used to erase data from the data fiel when it has been transferred to the VAX.
- Mode 5 - Option 2 of this mode is used to set the communication protocol used in mode 2 (see Appendix A for details).
- Mode 7 - On entering mode 7 one has three options which are again obtained by entering the appropriated number on the polycorder. Option 1 shows the clock/calendar, option 2 show the state of the batteries but option 3 is the most important - the terminal mode, Mode 7.3 is entered to send commands to the VAX especially logging in the "R U" - the command to run the data aquisition program.

TRANSFERRING DATA

Data is transferred from the polycorder to the VAX computer by means of an acoustically coupled modem. Acoustically coupled modems communicate through the phone system by having a telephone handset placed into rubber sockets in the modem. Modem and phone are not electrically connected but communicate sonically. In most instances this method is less

desirable than directly connected modems which plug straight into wall phone sockets - these can transmit at greater rates and are considered more reliable. However a prime consideration here was that the phones to be used might well not have jack plugs - such as those in motel rooms or mine offices.

To transfer data the line and column counter of the file to be sent is first set to zero. Next the handset is lifted and the number of the VAX's modem (247-2961) dialed. When the computer responds with a tone the handset is placed in the modem which has previously been connected to the serial port of the polycorder. The polycorder now should be put into terminal mode (7.3) where upon it should behave like a terminal; the enter key replacing the return key of a terminal. Thus the polycorder display should show parts of the VMS prompts for logging on (the usernamme and password, which must be entered via the polycorder keypad. Then the prompt "$" should appear (the DCL prompt); now the command "RUN U" should be given. This can be truncated to:

r [ENTER]

u [ENTER]

The computer will now run a small fortran program (U.FOR). Future communication is with this program. This program prompts you for the name of the file to which the data should go (at present the prompting is clumsy, many prompts do not remain on the LCD display). A name should now be entered on the polycorder keypad which will be the file in which the data will be put on the VAX. Having entered this the progrm will pause for a few seconds to allow the operator to prepare the polycorder for transmission. The polycorder is put into mode (2.2) ([ESC] 2 [ENTER] 2 [ENTER]). Now the file to be transmitted is selected by pressing the right or left facing arrows (->) or (<-) until drag.d appears on the LCD, then hit [ENTER].

At this stage the polycorder should be displaying various parameters associated with the data transmission. The L and C (line and column) numbers should be initially set to 1. If after a few seconds the line number does not start to increment a "V" should be entered on the polycorder keyboard which will start the transmission. When the L value ceases incrementing and has reached the line number of the last piece of data the polycorder should terminate the transmission with the message "XXX" which the fortran program will interpret as an end flag and terminate.

PROCESSING THE DATA

The polycorder transmits data in ASCII code. The resulting file created is a text file. This can be visually inspected and altered with the system editor (EDT). Files can be concatenated using system (DCL) commands.

The first fortran program ("crunch") operates on the file at low level. The file to be processed is taken line by line and then separated into component fields. Lines of text and lines which contain a zero event times are eliminated. Character data is then converted to real or integer data, and button numbers are converted to dig depths, swing angles and dump heights. The time lengths for each event are calculated by subtracting the time at the beginning from the time at the end of each event (using modulo 60 arithmetic). This program then writes out these times, depths, angles etc. to a target file for future processing. A copy of the results are printed on the terminal.

At this stage it is important to edit the results. The remaining function of the watch and notepad is to record inputting errors or events that are outside the capacity of the keypad to enter. The current version of crunch writes to file using an unformatted "write" statement. This allows them to be edited directly using EDT, the system editor.

The data base has now been created - from now on the problem is one of interpretation. However, it is an important first step to summarise the extent of the coverage. For each value of dig depth, swing angle, and drop height entered on the polycorder the mean and standard deviation of the dig time, swing time, drop time and return time are calculated by the program SUMM.FOR. Much further analysis is needed to interpret these results, however, it is outside the scope of this report.

BAD CHOICES - FUTURE IMPOVEMENTS

The following flaws were found in the system:

1. The "beeps" - the sound made by the polycorder were often not loud or high enough in pitch to be heard over the operating noise of the dragline.

2. The stored value representing an unpressed event button should not be zero (since this is a possible time value) but should be greater than 6000. On average one event in one thousand occurs "on the minute" and is missidentified as an unpressed button.
3. The transmission system is not perfect. Breaks occur which result in the file not being received. It is difficult to verify the transmission of a file using the polycorder as a remote terminal. The solution is probably to have the data transmitted by a specially written polycorder and VAX programs which greatly increase the amount of checking and verification. Fortunately transmission was only a problem during the day when staff were on hand in the office to check the files received.
4. The file pointers must be carefully checked and left in the proper position to avoid overwriting and misformatting.
5. The system is overly dependent on visually estimated variables. The addition of some visual reference markers in the field would greatly aid this estimation.

ADVANTAGES OF THIS SYSTEM

1. The design construction and programming of the system was started less than 2 weeks before the field work commenced. This involved the authors of this report only.
2. The additional equipment purchased consisted of $65 for the keyboard components and $100 for a second-hand modem only.
3. The amount of data gathered was large and it is believed reliable and accurate. The process of processing of this data was also greatly aided by having it transmitted directly onto computer file.

APPENDIX A - COMMUNICATIONS PARAMETERS

TERMINAL SETTINGS

Terminal: TXA0 Device Type: Unknown Owner: No Owner

Input: 300	LFfILL: 0	Width: 80	Parity: Even
Output: 300	CRfill: 0	Pase: 24	

Terminal Characteristics:

Interactive	Echo	Type-ahead	No Escape
Hostsync	TTsyn	Lowercase	Tab
Wrap	Scope	No Remote	No Holdscreen
No Eightbit	Broadcast	No Readsync	No Form
Fulldup	Modem	No Local-echo	No Autobaud
Hangup	No Brdcstmbx	DMA	No Altypeahd
Set-speed	No ANSI-CRT	no Regis	No Block-mode
No Advanced-video	No Edit-mode	No DEC-CRT	

POLYCORDER PROTOCOL

Baud-300
Parity-even
Full duplex
Transmit prompts-Y
Line req [CR]
Rep req-B
Cont req-V
Stop req-D
INP LINE TERM-[CR]
Max input line-80 char
Check sum-N
EOL [CR]
EOP-nothing

MODEM SETTINGS

-VAX modem must be set to LS (low speed), polycorder modem full duplex.

APPENDIX B - THE AUXILIARY KEYPAD KEY DESIGNATIONS

KEY NUMBERS (HARDWIRED)

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31

KEY FUNCTIONS (as defined by polycode program Drag.P.)

Dig depth	-20	-15	-10	-5	0	5	10	5
Swing angle	180	165	150	135	120	105	90	75
Drop height	5	10	15	20	25	30	35	40
Time Event	start dig	start swing	start dump	start retn	end cycle	unneeded buttons		

APPENDIX C - THE DATABASE - INITIAL COMPUTATIONS

The database exists in 2 forms and may be obtained in 3 formats:

1. The as received files - The daily transmission records have names such as DRAG1.DAT, DRAG2.DAT, etc.
2. These files have been composited into 4 files and then processed using CRUNCH.FOR to produce:

   CD1360.DAT - File containing data gathered on 30/7/84, 31/7/84 and 3/8/84. Thi is dayshift data from the 1360 dragline - one of whose swing motors was not working during this period.

   CD8050.DAT - Data gathered on 1/8/84 and 2/8/84 on the 8050 dragline working in hard ground - dayshift data.

   CN1360.DAT - Data gathered on the nightshifts of 9-10/8/84 and 10-11/8/84 on the 1360 dragline (motor repaired).

   CN8050.DAT - Nightshift data for 8050 dragline gathered on 7-8/8/84 and 8-9/8/84 shifts.

These "crunched" files may either be printed (and edited) directly or printed out using PR.FOR.

Example of "as received" file format

DIG	SWI	DUM	RET ESEC	EHM	D	S	H
5128	0	0	01598	1617	2	2	0
5715	1639	4722	54402792	1620	5	0	3
2944	5927	873	16715048	1621	5	0	3
5168	349	3562	43011690	1623	3	0	3
1848	3056	5942	5563929	1624	3	0	3
4136	5945	2032	2636 196	1626	3	0	3
337	2191	4774	55242829	1627	4	0	3
3012	3971	1246	15785304	1628	4	0	3
5569	706	3700	41591871	1630	5	0	3
1986	4073	1394	20625638	1631	5	0	3
5801	1829	5091	55723059	1633	5	0	3
3288	4399	1534	21775721	1634	4	0	3
5930	1779	4929	56492988	1636	4	0	3
3213	4166	1208	17845297	1637	4	0	3
5670	1607	4739	55602910	1639	4	0	3
3166	5614	2634	3510 729	1641	4	0	3
874	2508	5592	903415	1642	4	0	3
3613	5533	2679	3338 666	1644	4	1	3
850	2591	5811	2163472	1645	4	1	3
3607	5471	2488	3075 367	1647	4	1	3
827	2937	5970	4703784	1648	4	1	3
0	5784	2352	31015755	1649	4	2	3
5964	1888	4800	54282193	1651	4	3	3
2520	5045	1825	23685173	1652	4	3	3
5315	1311	4187	48221541	1654	4	3	3
5066	1085	3616	40801085	1656	4	3	3
1407	3576	107	8033778	1657	4	3	3
3916	2068	4558	51791745	1659	4	3	3
2000	3811	400	8993567	1700	4	3	3
3808	2211	4297	5014 250	1702	4	3	3
2986	4196	1669	21155846	1712	5	0	3
11	1389	4648	52732852	1714	5	0	3
3046	4549	1741	22555637	1715	4	0	3
5972	1693	5206	93298	1717	4	0	3
3506	5458	2643	32791011	1719	5	0	3
1172	3258	650	11764495	1720	4	0	3
4697	753	3812	43921935	1722	4	0	3
2065	4565	1596	21875871	1723	4	0	3

Example of "crunched" file format

22.80000	24.40000	4.710000	31.75000	2118.000
0	150	15		
21.34000	27.97000	3.860000	32.50000	2120.000
5	200	10		
19.19000	27.45000	3.700000	31.22000	2121.000
5	200	10		
25.57000	28.42000	3.130000	32.63000	2123.000
5	200	15		
25.13000	27.90000	2.740000	30.24000	2124.000
0	200	15		
16.06000	26.73000	5.030000	31.35000	2125.000
0	200	15		
13.19000	26.80000	4.780000	34.16000	2127.000
0	200	15		
23.45000	26.28000	4.570000	30.38000	2128.000
0	200	15		
23.34000	27.86000	3.530000	31.30000	2130.000
0	200	15		
25.53000	25.04000	4.470000	32.04000	2131.000
0	200	15		
23.47000	24.87000	3.970000	30.70000	2134.000
0	200	15		
45.52000	23.11000	4.320000	29.83000	2136.000
0	200	15		
11.95000	27.11000	2.370000	31.23000	2137.000
0	200	15		
23.24000	26.47000	3.720000	32.55000	2138.000
0	200	15		
19.71000	26.42000	3.390000	32.30000	2140.000
0	200	15		
25.70000	26.39000	3.580000	29.07000	2141.000
5	200	15		
13.95000	23.43000	4.550000	33.03000	2143.000
0	200	15		

Example of format using "PR.FOR"

```
$ r pr
enter name of file to be output
cn1360.dat
dis     swing   dump    retn    time    depth angle height
22.80   24.40    4.71   31.75   2118.      0   150     15
21.34   27.97    3.86   32.50   2120.      5   200     10
19.19   27.45    3.70   31.22   2121.      5   200     10
25.57   28.42    3.13   32.63   2123.      5   200     15
25.13   27.90    2.74   30.24   2124.      0   200     15
16.06   26.73    5.03   31.35   2125.      0   200     15
13.19   26.80    4.78   34.16   2127.      0   200     15
23.45   26.28    4.57   30.38   2128.      0   200     15
23.34   27.86    3.53   31.30   2130.      0   200     15
25.53   25.04    4.47   32.04   2131.      0   200     15
23.47   24.87    3.97   30.70   2134.      0   200     15
45.52   23.11    4.32   29.83   2136.      0   200     15
11.95   27.11    2.37   31.23   2137.      0   200     15
23.24   26.47    3.72   32.55   2138.      0   200     15
19.71   26.42    3.39   32.30   2140.      0   200     15
25.70   26.39    3.58   29.07   2141.      5   200     15
13.95   23.43    4.55   33.03   2143.      0   200     15
28.77   26.80    3.87   37.64   2148.      0   200     15
17.63   31.95    4.03   32.83   2150.      0   200     15
22.76   28.09    5.35   31.31   2212.      0   200     15
25.13   29.31    3.20   34.80   2214.      0   200     15
29.72   26.79    4.22   29.09   2215.      5   200     15
12.73   25.98    5.76   30.21   2216.      0   200     15
14.01   24.65    5.09   33.94   2218.      0   200     15
41.87   28.41    3.78   29.04   2220.      0   200     15
49.66   28.79    5.08   32.20   2222.      0   200     20
21.80   26.17    5.05   36.19   2223.      5   200     20
20.50   25.99    5.94   30.91   2225.      5   200     20
19.22   28.26    3.33   31.98   2226.      5   200     20
15.70   31.67    5.01   31.18   2230.      5   200     20
22.65   29.68    3.35   31.07   2231.      5   200     20
21.90   26.71    3.44   30.91   2233.      5   200     20
24.17   28.27    4.38   30.76   2234.      0   200     20
21.80   26.21    4.90   30.21   2235.      0   200     20
18.13   27.38    3.69   32.25   2237.      0   200     20
22.16   25.73    4.04   33.11   2238.      0   200     20
21.96   28.22    5.40   30.67   2240.      0   200     20
24.12   27.14    4.42   30.88   2241.      0   200     20
21.69   27.44    3.88   27.81   2243.      0   200     20
15.91   27.08    3.72   31.00   2244.      0   200     20
21.52   28.79    4.18   33.40   2247.      0   200     20
23.33   27.60    3.65   30.76   2249.      0   200     20
14.30   25.71    4.02   32.12   2250.      0   200     20
21.38   25.24    6.06   28.05   2251.      0   200     20
15.57   25.16    4.39   31.42   2253.      0   200     20
13.85   25.31    5.71   25.94   2254.      0   200     20
22.33   27.04    3.96   31.96   2256.      0   200     20
```

Program "PR.FOR" for printing database files

```
      program pr
c     this program prints out the data files in tabular form.

      real dig, swing, dump, retn, cycle, time
      integer depth, angle, height
      character*10 filename

      type*, 'enter name of file to be output'
      read 110, filename
110   format (a)
      type*,'dig    swing  dump   retn   time   depth angle height'
      open (unit=1,file=filename,status='unknown')

220   read (1,*,end=1111) dig, swing, dump, retn, time
      read (1,*) depth, angle, height
      type 111, dig,  swing, dump, retn, time, depth, angle, height
111   format(' ',f5.2,'  ',f5.2,'  ',f5.2,'  ',f5.2,'  ',f5.0,'  ',I4,'  ',
     5 I4,'  ',I4)
      goto 220
1111  close (unit=1)
      end
```

Program "crunch" used to compute times etc. from file received from polycorder

```
      program crunch

c     This is intended to read the files transmitted by the polycorder,
c     break them down into component fields and work out the times for
c     each event.


      character*10 filename,filename2,inline*80, ssdig*4, ssswing*4, ssdump*4
      character ssretn*4, seoc*4, stim*4, sdepth*1, sangle*1, sheight*1
      integer depth, angle, height, count
      integer sdig, sswing, sdump, sretn, eoc, tim
      real dig, swing, dump, retn, cycle, time
      logical ok, zero


      type*, 'Enter the file name to be processed:'
      read 110, filename
      type*, 'Enter the Output file name:'
      read 110, filename2

110   format (a)

      open (unit=1,file=filename,status='unknown')
      open (unit=2,file=filename2,status='new')
      count = 0
500   continue
      count = count + 1
      read (1,110,end = 1000), inline

      zero = .false.
      ok = .true.
```

```
c       pick inline apart as substrings
        ssdig = inline(4:7)
        ssswing = inline(10:13)
        ssdump = inline(16:19)
        ssretn = inline(22:25)
        seoc = inline(26:29)
        stim = inline(32:35)
        sdepth = inline(38:38)
        sangle = inline(41:41)
        sheight = inline(44:44)

c       calculate button numbers
        angle =( ichar(sangle))-48
        height = ichar(sheight)-48
        depth = ichar(sdepth)-48

c       Use subroutine 'num' to convert to digits and set 'ok' true if chars'
        call num (ok, ssdig, sdig)
        call num (ok, ssswing, sswing)
        call num (ok, ssdump, sdump)
        call num (ok, ssretn, sretn)
        call num (ok, seoc, eoc)
        call num (ok, stim, tim)


c       calculate differences (event times) in hunderdths of secs
        dig = (sswing - sdig)
        swing = (sdump - sswing)
        dump = (sretn - sdump)
        retn = ( eoc - sretn)
        time = tim

c       correct if minute value changes
        if (swing .lt. 0) swing = swing + 6000
        if (dig .lt. 0) dig = dig + 6000
        if (dump .lt. 0) dump = dump + 6000
        if (retn .lt. 0) retn = retn + 6000
```

```
c       divide by 100
        dig = dig/100
        swing = swing/100
        dump = dump/100
        retn = retn/100

c       calculate total cycle time
        cycle = swing + dig + dump + retn

c       run traps on zero values in event times
        zero = (( sswing .eq. 0) .or. (sdig .eq. 0) .or. (sdump .eq. 0))
        zero = ((zero) .or. (sretn .eq. 0))
        if (.not.(ok)) goto 999

        depth = (depth - 4) * 5
        angle = 180 - (angle * 15)
        height = (height + 1) * 5

c       print out the stuff

        if (zero) then
           print*, '***********caution incomplete button entry above'
        else
           type*, dig, swing, dump, retn, time
           type*,'            ',depth,angle,height
           write (2,*) dig,swing,dump,retn,time,depth,angle,height
        endif
        goto 500

999     type*, inline , 'Interpreted as character input'
        goto 500

1000    type*, ' ------Thats all folks'
        close (unit=2)
        end
```

```
subroutine num(flag, word, number)
character letter
character *4 word
integer m, i, k
number = 0
m=1
do i = 4,1,-1
letter = word(i:i)
if (letter .eq. ' ') letter = '0'
k = ichar(letter) - 48
if (( k .lt. 0) .or. (k .gt. 9)) flag = .false.
number = number + (m * k)
m = m * 10
end do
end
```

APPENDIX D - FORTRAN PROGRAM "U" FOR TRANSFERRING DATA ONTO VAX

```
      character       filename*10,inline*80
      type*,'enter filename on Vax'
      read 110,  filename
110   format (a)
      open (unit=1,file=filename,status='unknown')

      do i=1,2000000
          k = i
      end do

      print*,'V'

111   read (5,110) inline
      if (inline .eq. ' end') goto 120
      write(1,*),inline
      goto 111

120   type*,'file accepted'
      end
```

APPENDIX E - POLYCODE PROGRAM FOR DATA ACQUISITION

```
PGM
DRAG.P
  1.   31   OPN   DRAG.D
  2.   51   IN9
  3.    3   JNZ   2
  4.   26   CON   255
  5.   50   INP
  6.   80   SUB
  7.   14   STO   0
  8.   26   CON   7
  9.   80   SUB
 10.    6   JGZ   15
 11.   15   RCL   0
 12.   14   STO   1
 13.   22   SNG   0,25
 14.    1   JMP   104
 15.   26   CON   8
 16.   80   SUB
 17.    6   JGZ   22
 18.   15   RCL   0
 19.   14   STO   2
 20.   22   SNG   15,25
 21.    1   JMP   104
 22.   26   CON   8
 23.   80   SUB
 24.    6   JGZ   29
 25.   15   RCL   0
 26.   14   STO   3
 27.   22   SNG   30,25
 28.    1   JMP   104
 29.   66   TIM
 30.   14   STO   4
 31.   76   POP
 32.   14   STO   5
 33.   15   RCL   0
 34.   26   CON   24
 35.   80   SUB
 36.    6   JGZ   41
 37.   15   RCL   4
 38.   14   STO   6
 39.   22   SNG   90,05
 40.    1   JMP   104
 41.   26   CON   1
 42.   80   SUB
 43.    6   JGZ   49
 44.   15   RCL   4
 45.   14   STO   8
 46.   22   SNG   70,10
 47.   22   SNG   80,10
 48.    1   JMP   104
 49.   26   CON   1
 50.   80   SUB
 51.    6   JGZ   57
 52.   15   RCL   4
 53.   14   STO   10
 54.   22   SNG   20,10
 55.   22   SNG   45,10
 56.    1   JMP   104
 57.   26   CON   1
 58.   80   SUB
 59.    6   JGZ   65
 60.   15   RCL   4
 61.   14   STO   12
 62.   22   SNG   60,10
 63.   22   SNG   50,10
 64.    1   JMP   104
 65.   15   RCL   6
 66.   77   STF
 67.   61   ICP
 68.   15   RCL   8
 69.   77   STF
 70.   61   ICP
 71.   15   RCL   10
 72.   77   STF
 73.   61   ICP
 74.   15   RCL   12
 75.   77   STF
 76.   61   ICP
 77.   15   RCL   4
 78.   77   STF
 79.   61   ICP
 80.   15   RCL   5
 81.   77   STF
 82.   61   ICP
 83.   15   RCL   1
 84.   77   STF
 85.   61   ICP
 86.   15   RCL   2
 87.   26   CON   8
 88.   80   SUB
 89.   77   STF
 90.   61   ICP
 91.   15   RCL   3
 92.   26   CON   16
 93.   80   SUB
 94.   77   STF
 95.   61   ICP
 96.   26   CON   0
 97.   14   STO   6
 98.   14   STO   8
 99.   14   STO   10
100.   14   STO   12
101.   22   SNG   20,10
102.   22   SNG   40,10
103.   22   SNG   50,20
104.   51   IN9
105.    2   JPZ   104
106.   20   CDS   0,32
107.   65   MEM
108.   37   VUA
109.    1   JMP   2
110.    0   END   *
111.    0   END   *
```

APPENDIX F - THE DAILY FIELD ROUTINE

1. On reaching the site: connect keyboard to polycorder.
2. turn polycorder on and clear data file by use of mode 4.
3. Check and set datafile pointers (mode 1).
4. Run keyboard program (mode 0) program: DRAG.P
5. Enter data using keyboard.
6. Exit from program.
7. On reaching convenient location with phone: connect polycorder to modem using a 25 pin cable.
8. Dial the Calgary CRL number (403-247-2961) and wait for tone.
9. Place handset on modem.
10. Turn polycorder onto mode 7:3
11. Log into VAX.
12. Run VAX unloading program (NU.EXE).
13. Goto mode 2:1 or 2:2 on polycorder.
14. Type "V" to start transmission.
15. When transmission is over goto polycroder mode 7:3 and log out.
16. If required use mode 4 to clear the data file and then reset pointers.
17. At a convenient location plug the polycorder into a recharger.

APPENDIX G - SUMMARY OF DATA - THE INITIAL ANALYSIS

This appendix contains: (a) A listing of SUMM.FOR - the program used to summarize the data.

(b) The summary of all 4 data files.

```
      program sum2
c     this program prints out a summary of the data in each file

      real tdat(5,2000),mean, sd
      integer idat(3,2000),count,scan,i,j, size,loop
      character*10 filename

      count = 1
c     read in data from file
      type*, 'enter name of file to be output'
      read 110, filename
110   format (a)
      open (unit=1,file=filename,status='unknown')

220   read (1,*,end=1111) (tdat(i,count), i=1,5)
      read (1,*) (idat(i,count), i = 1,3)
      count = count +1
      goto 220
1111  close (unit=1)
      size = count - 1
```

```
c       compute summaries of data

        type*, 'Basic Stats:'


c       Scan the three integer columns with i, chose the right t to compare
        do 340 i=1,3
c           scan the time data matrix
            do 341 j=1,4
c                   set up parameters for looping through values
                if (i .eq. 1) then
                    type*, 'THE EFFECT OF DIG DEPTH.....'
                    low = -80
                    high = 55
                    incr = 5
                end if

                if (i .eq. 2) then
                    type*, 'THE EFFECT OF SWING ANGLE...'
                    low = 360
                    high = 0
                    incr = -5
                end if

                if (i .eq. 3) then
                    type*, 'THE EFFECT OF DUMP HEIGHT...'
                    low = -45
                    high = 80
                    incr = 5
                end if

                if (j .eq. 1) type*, '.....on the dig time'
                if (j .eq. 2) type*, '.....on the swing time'
                if (j .eq. 3) type*, '.....on the dump time'
                if (j .eq. 4) type*, '.....on the return time'
```

```
c               scan specific values
                type*
                type*, '      class   no.of data  mean            st.dev'
                do 342 loop = low, high, incr
                    count = 0
                    sd = 0
                    mean = 0
                    do 343 scan = 1, size
c                       scan data matrix lookins for specific values -keep totals
                        if (loop .eq. idat(i,scan)) then
                            mean = mean + tdat(J,scan)
                            sd = sd + ((tdat(J,scan))*(tdat(J,scan)))
                            count = count+ 1
                        end if
343                 continue

c               calc stats
                    if (count .ne. 0) then
                        if (count .eq. 1) then
                            sd = 0
                        else
                            sd = sqrt((sd - mean*mean/count)/(count - 1))
                        end if
                        mean = mean/count
                        type*, loop, count, mean, sd
                    end if
342             continue
                type*

341         continue
340     continue


        end
```

File CD 1360.DAT - file of daytime data from dragline 1360

cd1360.dat
Basic Stats:
THE EFFECT OF DIG DEPTH.....
.....on the dig time

class	no.of data	mean	st.dev
-15	7	26.11571	9.774338
-10	12	22.55167	5.474514
-5	3	13.99333	3.550380
0	316	19.69690	9.160084
5	135	17.24971	6.759508
10	17	16.68412	4.140540

THE EFFECT OF DIG DEPTH.....
.....on the swing time

class	no.of data	mean	st.dev
-15	7	17.30143	1.757668
-10	12	19.34500	3.518979
-5	3	27.28667	5.792535
0	316	24.50946	5.314830
5	135	25.95089	4.487025
10	17	25.97882	3.847852

THE EFFECT OF DIG DEPTH.....
.....on the dump time

class	no.of data	mean	st.dev
-15	7	5.228572	1.933126
-10	12	4.265833	1.670468
-5	3	6.523333	0.7522219
0	316	4.958073	1.510551
5	135	5.341852	1.550496
10	17	5.207059	0.9693533

THE EFFECT OF DIG DEPTH.....
.....on the return time

class	no.of data	mean	st.dev
-15	7	20.59714	3.200609
-10	12	21.44750	3.978134
-5	3	34.40667	1.036563
0	316	26.04774	5.994595
5	135	26.82866	4.827571
10	17	26.58529	4.961983

THE EFFECT OF SWING ANGLE...
.....on the dig time

class	no.of data	mean	st.dev
220	36	22.24583	9.205443
200	64	19.62172	8.253184
180	35	16.31286	5.504317
165	7	18.71143	4.313642
150	41	18.71049	6.359678
135	118	18.53094	6.951572
120	74	20.86568	10.51162
105	53	14.96434	7.895084
90	44	20.25296	10.64474
75	18	21.74000	8.163399

THE EFFECT OF SWING ANGLE...
.....on the swing time

class	no.of data	mean	st.dev
220	36	30.22611	2.526467
200	64	29.87500	2.832075
180	35	29.74629	4.736011
165	7	28.58714	3.147562
150	41	25.37634	4.103680
135	118	24.48162	3.381572
120	74	22.17378	3.265579
105	53	21.94622	4.776318
90	44	19.31818	2.354380
75	18	16.69278	2.967685

THE EFFECT OF SWING ANGLE...
.....on the dump time

class	no.of data	mean	st.dev
220	36	5.094167	1.533998
200	64	5.421562	1.344468
180	35	5.787142	1.460753
165	7	4.880000	1.224637
150	41	4.946342	1.500033
135	118	4.881442	1.539038
120	74	5.062432	1.809103
105	53	4.898868	1.457742
90	44	4.875454	1.244602
75	18	4.950556	1.595626

THE EFFECT OF SWING ANGLE...
.....on the return time

class	no.of data	mean	st.dev
220	36	32.18139	5.322983
200	64	31.48735	3.259582
180	35	33.81286	2.981619
165	7	33.26572	2.084479
150	41	25.80439	3.564131
135	118	24.95433	4.050175
120	74	23.97905	4.333642
105	53	22.61925	3.161452
90	44	20.43273	2.610964
75	18	19.15278	3.277992

THE EFFECT OF DUMP HEIGHT...
.....on the dig time

class	no.of data	mean	st.dev
5	35	25.11200	7.296114
10	164	17.38927	6.742942
15	251	19.30172	9.367424
20	40	18.91200	7.303905

THE EFFECT OF DUMP HEIGHT...
.....on the swing time

class	no.of data	mean	st.dev
5	35	21.52029	4.204100
10	164	22.78592	4.202727
15	251	25.69658	5.115674
20	40	29.62925	4.679298

THE EFFECT OF DUMP HEIGHT...
.....on the dump time

class	no.of data	mean	st.dev
5	35	4.517715	1.709532
10	164	5.111220	1.510086
15	251	4.976536	1.495453
20	40	5.957751	1.279848

THE EFFECT OF DUMP HEIGHT...
.....on the return time

class	no.of data	mean	st.dev
5	35	22.65800	3.716641
10	164	23.90018	4.339967
15	251	27.02701	5.735618
20	40	32.83099	4.610044

File CD 8050.DAT - file of daytime data from dragline 8050

cd8050.dat
Basic Stats:
THE EFFECT OF DIG DEPTH.....
.....on the dig time

class	no.of data	mean	st.dev
-10	41	22.34781	7.749899
-5	108	19.64842	8.424263
0	95	19.26695	10.41669

THE EFFECT OF DIG DEPTH.....
.....on the swing time

class	no.of data	mean	st.dev
-10	41	24.32610	5.850840
-5	108	19.94972	5.092235
0	95	19.04464	3.992581

THE EFFECT OF DIG DEPTH.....
.....on the dump time

class	no.of data	mean	st.dev
-10	41	4.977560	1.180896
-5	108	4.769167	1.189496
0	95	4.705894	1.863341

THE EFFECT OF DIG DEPTH.....
.....on the return time

class	no.of data	mean	st.dev
-10	41	22.43439	7.253932
-5	108	17.33472	6.393327
0	95	17.08389	5.345830

THE EFFECT OF SWING ANGLE...
.....on the dig time

class	no.of data	mean	st.dev
240	47	21.35936	8.391071
180	8	13.73000	4.130834
165	3	16.19333	3.254265
120	18	21.30111	8.892546
105	18	21.80834	8.718777
90	107	21.28336	10.40327
75	43	15.18721	5.545203

THE EFFECT OF SWING ANGLE...
.....on the swing time

class	no.of data	mean	st.dev
240	47	29.26937	2.626682
180	8	25.29250	2.344717
165	3	21.90667	0.5052218
120	18	19.05167	1.644843
105	18	17.30778	4.231092
90	107	18.41421	1.823537
75	43	16.10861	1.521608

THE EFFECT OF SWING ANGLE...
.....on the dump time

class	no.of data	mean	st.dev
240	47	5.347234	2.540622
180	8	5.553751	0.8905663
165	3	4.373333	2.492596
120	18	4.948889	1.289693
105	18	4.446112	0.8541662
90	107	4.471496	0.9499566
75	43	4.878604	1.101056

THE EFFECT OF SWING ANGLE...
.....on the return time

class	no.of data	mean	st.dev
240	47	29.14298	3.847609
180	8	23.02625	1.872152
165	3	20.48667	4.004568
120	18	15.63778	4.226535
105	18	13.90778	2.406659
90	107	15.24570	2.406441
75	43	14.80070	3.887302

THE EFFECT OF DUMP HEIGHT...
.....on the dis time

class	no.of data	mean	st.dev
5	8	27.43750	15.28353
10	29	17.29724	5.505096
15	21	20.48524	5.956803
20	184	20.05271	9.508986
25	2	13.82000	3.097122

THE EFFECT OF DUMP HEIGHT...
.....on the swing time

class	no.of data	mean	st.dev
5	8	31.81625	3.141974
10	29	27.33517	3.940121
15	21	26.13667	4.716318
20	184	18.11831	2.755223
25	2	15.64500	1.096008

THE EFFECT OF DUMP HEIGHT...
.....on the dump time

class	no.of data	mean	st.dev
5	8	6.247500	1.897168
10	29	5.588621	2.823874
15	21	4.807619	1.425130
20	184	4.576032	1.048967
25	2	5.605000	4.9526259E-02

THE EFFECT OF DUMP HEIGHT...
.....on the return time

class	no.of data	mean	st.dev
5	8	31.26500	5.134172
10	29	26.52345	4.585361
15	21	26.71334	5.301717
20	184	15.27903	3.475882
25	2	11.65500	0.6010296

File CN 1360.DAT - file of nighttime data from dragline 1360

cn1360.dat
Basic Stats:
THE EFFECT OF DIG DEPTH.....
.....on the dig time

class	no.of data	mean	st.dev
0	148	19.89337	8.612537
5	94	16.82266	5.378900
10	6	17.12000	4.567591

THE EFFECT OF DIG DEPTH.....
.....on the swing time

class	no.of data	mean	st.dev
0	148	25.78136	1.913397
5	94	24.81032	1.987317
10	6	26.19167	1.721447

THE EFFECT OF DIG DEPTH.....
.....on the dump time

class	no.of data	mean	st.dev
0	148	3.855744	1.173269
5	94	3.738086	1.032425
10	6	3.068333	0.5417907

THE EFFECT OF DIG DEPTH.....
.....on the return time

class	no.of data	mean	st.dev
0	148	30.51736	2.461912
5	94	29.58170	2.997815
10	6	31.65333	2.633190

THE EFFECT OF SWING ANGLE...
.....on the dig time

class	no.of data	mean	st.dev
200	68	22.02485	7.980247
180	66	16.86954	6.376680
165	61	17.30590	8.343340
150	45	17.88444	6.597936
135	8	19.59125	4.246055

THE EFFECT OF SWING ANGLE...
.....on the swing time

class	no.of data	mean	st.dev
200	68	27.21485	1.675051
180	66	25.57788	1.552816
165	61	24.46705	1.631657
150	45	24.07977	1.380284
135	8	23.76625	1.258648

THE EFFECT OF SWING ANGLE...
.....on the dump time

class	no.of data	mean	st.dev
200	68	4.305882	1.117298
180	66	3.385151	0.9185137
165	61	3.862787	0.9864489
150	45	3.646888	1.244885
135	8	3.060000	1.034850

THE EFFECT OF SWING ANGLE...
.....on the return time

class	no.of data	mean	st.dev
200	68	31.69985	2.046737
180	66	30.83288	2.536406
165	61	29.48967	2.438755
150	45	28.45511	2.756900
135	8	27.15750	1.389534

THE EFFECT OF DUMP HEIGHT...
.....on the dig time

class	no.of data	mean	st.dev
10	111	18.26351	7.782736
15	76	18.65066	7.210109
20	61	19.40279	7.791764

THE EFFECT OF DUMP HEIGHT...
.....on the swing time

class	no.of data	mean	st.dev
10	111	25.06405	1.693487
15	76	25.24895	2.036784
20	61	26.29394	2.193593

THE EFFECT OF DUMP HEIGHT...
.....on the dump time

class	no.of data	mean	st.dev
10	111	3.566667	1.046220
15	76	3.757632	1.026024
20	61	4.245246	1.220369

THE EFFECT OF DUMP HEIGHT...
.....on the return time

class	no.of data	mean	st.dev
10	111	30.34027	2.683943
15	76	30.11210	2.718647
20	61	30.01443	2.808632

File CN 8050.DAT - file of nighttime data from dragline 8050

cn8050.dat
Basic Stats:
THE EFFECT OF DIG DEPTH.....
.....on the dig time

class	no.of data	mean	st.dev
-15	2	26.69000	11.31370
-10	108	18.92861	4.575939
-5	113	21.97141	8.689025
0	71	22.90605	15.48331

THE EFFECT OF DIG DEPTH.....
.....on the swing time

class	no.of data	mean	st.dev
-15	2	18.75000	1.173800
-10	108	18.71083	2.878870
-5	113	18.71221	3.292519
0	71	20.75197	10.19824

THE EFFECT OF DIG DEPTH.....
.....on the dump time

class	no.of data	mean	st.dev
-15	2	4.300000	0.0000000E+00
-10	108	3.903055	1.045114
-5	113	4.597700	1.310892
0	71	4.951689	1.761626

THE EFFECT OF DIG DEPTH.....
.....on the return time

class	no.of data	mean	st.dev
-15	2	18.13500	1.096036
-10	108	17.54509	3.457598
-5	113	17.38902	4.324183
0	71	20.47634	11.92024

THE EFFECT OF SWING ANGLE...
.....on the dis time

class	no.of data	mean	st.dev
320	9	16.46556	3.235224
135	7	20.90857	3.923462
120	19	18.88895	4.997297
105	32	15.25187	4.987181
90	94	20.62500	7.469652
75	31	22.03871	9.127413
55	22	21.89773	8.108155
50	1	29.38000	0.0000000E+00
45	6	15.08667	6.719033
35	73	25.22836	14.35199

THE EFFECT OF SWING ANGLE...
.....on the swing time

class	no.of data	mean	st.dev
320	9	46.29445	1.341477
135	7	26.03000	2.804198
120	19	21.86684	1.570690
105	32	21.14187	1.724760
90	94	18.20085	2.450943
75	31	18.02484	2.182077
55	22	18.19727	1.121097
50	1	18.87000	0.0000000E+00
45	6	16.70667	1.827911
35	73	15.97493	2.685243

THE EFFECT OF SWING ANGLE...
.....on the dump time

class	no.of data	mean	st.dev
320	9	4.897779	1.088433
135	7	2.530000	0.4138836
120	19	3.585790	1.188364
105	32	3.498438	0.7113631
90	94	4.121276	1.125345
75	31	4.143225	1.060550
55	22	4.981819	0.6732600
50	1	5.550000	0.0000000E+00
45	6	5.116667	0.9118028
35	73	5.447670	1.649256

THE EFFECT OF SWING ANGLE...
.....on the return time

class	no.of data	mean	st.dev
320	9	49.30111	1.155616
135	7	25.07429	6.259680
120	19	22.18895	2.944068
105	32	20.27219	2.867990
90	94	16.67883	3.167460
75	31	17.53032	3.001052
55	22	17.17864	2.599472
50	1	19.73000	0.0000000E+00
45	6	16.90167	2.240883
35	73	14.38452	3.683641

THE EFFECT OF DUMP HEIGHT...
.....on the dig time

class	no.of data	mean	st.dev
5	23	32.12739	22.55403
10	18	24.70834	8.401140
15	64	20.39797	6.484212
20	118	20.75432	7.297415
25	71	17.86774	6.975389

THE EFFECT OF DUMP HEIGHT...
.....on the swing time

class	no.of data	mean	st.dev
5	23	15.96391	3.692910
10	18	19.72222	10.72903
15	64	19.85938	9.301232
20	118	17.73119	2.109966
25	71	21.98155	2.074102

THE EFFECT OF DUMP HEIGHT...
.....on the dump time

class	no.of data	mean	st.dev
5	23	5.929132	2.193845
10	18	6.253334	1.353140
15	64	4.871719	0.8442333
20	118	4.194491	1.094720
25	71	3.458732	0.9213547

THE EFFECT OF DUMP HEIGHT...
.....on the return time

class	no.of data	mean	st.dev
5	23	15.48304	4.989760
10	18	17.47111	12.01064
15	64	18.90187	11.08014
20	118	16.57364	2.882617
25	71	21.32281	3.560628

POOR

QUALITY

ORIGINAL